Advanced Encryption Standard(AES)

Ashwini Kumar(2018MT60778)

The Advanced Encryption Standard (AES) is a block cipher selected to safeguard sensitive information by the U.S. government. To encrypt confidential data, AES is implemented in software and hardware around the world.
It is important for the security of government servers, cybersecurity and protection of electronic data.AES was first developed in 1997 by the National Institute of Standards and Technology (NIST) when it declared the need for an alternative to the Data Encryption Standard (DES), which was beginning to become vulnerable to brute-force attacks.

Three block ciphers are used in AES: AES-128, AES-192 and AES-256. In order to encrypt and decrypt a block of messages, AES-128 uses a 128-bit key length, while AES-192 uses a 192-bit key length and AES-256 a 256-bit key length for symmetric encryption. Using cryptographic keys of 128, 192 and 256 bits respectively, each cipher encrypts and decrypts data in blocks of 128 bits. For 128-bit keys, there are 10 rounds, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys. A round consists of several process stages that involve the substitution, transposition and mixing of the plaintext input to translate it into the ciphertext final output.

SECURITY/VULNERABILITIES IN AES:

In today's cryptography, in both hardware and software, AES is commonly adopted and supported. No functional cryptanalytic attacks against AES have been found to date. In addition, AES has built-in key duration versatility, which provides a degree of 'future-proofing' toward improvement in the ability to execute extensive key searches.

The 256-bit key is the longest, offering the highest degree of encryption. With a 256-bit key, to ensure the correct one is used, a hacker will need to try 2256 different combinations. This number is astronomically high, landing at a total of 78 digits. It is infinitely greater than the observable universe's number of atoms. Understandably, the US government requires 128- or 256-bit encryption for sensitive data.

The number of rounds of encryption also separates the three AES varieties. AES 128 uses 10 rounds, 12 rounds are used by AES 192, and 14 rounds are used by AES 256. The more rounds, the more difficult the encryption, which makes AES 256 the most reliable implementation of AES.This should be observed that with a longer key and more rounds comes higher performance requirements. AES 256 uses 40 percent more device resources than AES 192, so it is better suited to environments of high sensitivity where protection is more critical than speed.

Is it possible to do cryptanalysis on AES?

By using brute-force methods, AES 256 is practically unbreakable. Although a 56-bit DES key can be cracked in less than a day, using current computing technologies, AES will take billions of years to break.
The key risk comes from side-channel attacks, because the AES cipher itself is so safe. These do not attempt a brute-force attack, but rather attempt to collect data that is escaping from the device. To try to figure out how the protection algorithms work, hackers can listen to sounds, electromagnetic signals, timing information, or power consumption. Side-channel attacks can be avoided by eliminating leaks of information or by masking leaked data (by producing additional electromagnetic signals or sounds) so that no useful information is generated. These side-channel threats can be guarded against by cautious implementation of AES.

Overall, AES is considered secure against brute-force attacks by security experts, where all possible key combinations are tested before the right key is identified. The key size used for encryption, however, must be big enough so that modern computers will not crack it, even considering improvements in processing speeds based on Moore's law. For brute-force attacks, a 256-bit encryption key is much harder to guess than a 128-bit key; but because the latter takes so long to guess, even with a huge amount of computing power, it is unlikely to be a problem for the foreseeable future, as a hacker would need to use quantum computing to generate the necessary brute force.

Since the standard was finalised in 2000, research into attacks on AES encryption has continued. Attacks against reduced-round versions of AES have been published by different researchers. Security experts maintain that when properly implemented, AES is secure.

A big challenge to AES encryption comes from side-channel attacks. Side-channel attacks are targeted at picking up leaked information from the device instead of attempting a brute-force assault. However, side-channel attacks can decrease the number of possible combinations needed with brute force to attack AES.

AES vs. RSA

RSA is an asymmetric cryptography system, unlike AES, which uses symmetric encryption. In order to encrypt and decrypt it, symmetric encryption requires converting plaintext to ciphertext using the same key or secret key. The word asymmetric, on the other hand, derives from the fact that two related keys are used for encryption: a public and a private key. If encryption is achieved with the public key, decryption can only occur with the private key associated with it, and vice versa. When there are two distinct endpoints, RSA keys are usually used.

While RSA encryption works well to safeguard data transmission across geographical boundaries, its performance is poor. The solution is to combine RSA encryption and AES encryption in order to take advantage of RSA security with AES efficiency. By generating a temporary AES key and securing it with RSA encryption, this can be done.

AES vs. DES

Symmetric block ciphers are both the two norms, but AES is more mathematically efficient. AES' main advantage lies in its key options for duration. The time required to crack an encryption algorithm is directly related to 128-bit, 192-bit or 256-bit keys, the length of the key used to secure communication. Therefore, AES is exponentially stronger than the 56-bit key of DES. AES encryption is also significantly faster, so it is ideal for applications, firmware and hardware that require low latency or high throughput.

HASHING FUNCTIONS:

Hash functions take as input a potentially long message and produce from the content a unique output value. A hash function's output is generally referred to as the message digest.

Hashing is a one-way function and there's no way to reverse the hashing process to expose the original input with a properly built algorithm. Hash functions are expected to generate the same output for the same input in the sense of digital signatures (deterministic). This helps a message receiver to recompute the message digest with the same hash function and compare it with the transmitted digest to check that the message in transit has not been altered.

SHA - Secure Hash Algorithm:

SHA-1 takes an input of practically any length and generates a response of 160 bits. It processes a message into blocks of 512 bits. The SHA algorithm pads the message with data until the length exceeds the next highest multiple of 512-bit if the message length is not a multiple of 512-bit.

SHA-1 against well-funded adversaries is no longer considered secure. In 2017, all major web browser manufacturers approved SHA-1 SSL certificates. In SHA-1, Google also showed a collision.

As a response to the deficiencies in SHA-1, SHA-2 was released in 2001. It requires substantial modifications from its predecessor and has four primary variants:

- Using a 512-bit block size, SHA-256 generates a 256-bit message digest.
- SHA-224 uses a truncated version of the SHA-256 hash and uses a 512-bit block size to generate a 224-bit digest.
- Using a 1,024-bit block size, SHA-512 generates a 512-bit message digest.
- SHA-384 utilises a truncated version of the SHA-512 hash and uses a 1,024-bit block size to generate a 384-bit digest.

The cryptographic community generally considers SHA-2 algorithms as secure, but it theoretically suffers from the same weakness as the SHA-1 algorithm.

SHA-3 was released in 2015, though SHA-3 is internally distinct from MD5 as a structure of SHA-1 and SHA-2 as part of the same set of standards. SHA-3 is a subset of Keccak's wider primitive cryptographic family. It was developed using a more stable algorithm as a drop-in replacement for SHA-2, providing the same variants (SHA3-256/SHA3-224/SHA3-512/SHA3-384) and hash lengths.

MD2 - Message Digest:

To provide a safe hash function for 8-bit processors, Ronald Rivest (yes, the one from Rivest, Shamir, and Adleman aka RSA Security) created the MD2 Message-Digest Algorithm in 1989.

MD2 pads the message to a length of a 16-bit multiple and calculates a checksum of 16 bytes(!) appended to the end of the input message. By using the original message along with the appended checksum, a 128-bit message digest is then created.

There are cryptanalytic attacks against the MD2 algorithm and it has also been demonstrated that MD2 is not a one-way function.

MD4 was implemented in 1990 and supports 32-bit processors as an upgrade to MD2. In an improved algorithm, this enhances the security level. In the MD4 algorithm, multiple bugs have been identified and it is now no longer considered reliable. If necessary, use should be avoided.

MD5 was published as the next version of the message digest algorithm in 1991. It also processes 512-bit message blocks but uses four computation rounds to create the same 128-bit digest length of the message as in MD2 and MD4. Recent cryptanalytic attacks have shown that MD5 is prone to collisions.

All algorithms are no longer recognised as acceptable hashing functions in the MD family. They can still be seen in use today, however.